N88-16439

APPLICATION OF ARTIFICIAL INTELLIGENCE TO IMPULSIVE ORBITAL TRANSFERS

by

Rowland E. Burns

NASA
Marshall Space Flight Center
Huntsville, AL

ABSTRACT

A generalized technique for the numerical solution of any given class of problems is presented. The technique requires the analytic (or numerical) solution of every applicable equation for all variables which appear in the problem. Conditional blocks are employed to rapidly expand the set of known variables from a minimum of input. The method is illustrated via the use of the Hohmann transfer problem from orbital mechanics.

INTRODUCTION

Although many papers deal with the use of rule based systems, few have applied that logic to strictly mathematical systems. Mathematics programs in artificial intelligence tend to provide the solution to a very specific problem such as evaluation of an integral.

The techniques presented here are for a very different class of problem. The user requires the numerical solution to an extended problem that could involve any number of equations from differing fields. While some of these equations are critical, others are inapplicable to the specific problem at hand; some of the critical equations are usable only after other equations are applied in a specific order. The non-expert cannot be expected to know all of these equations, the order in which they must be employed, or even whether or not there is enough information at hand to solve the assigned problem. This paper suggests a method to eliminate such difficulties. The method has been reported in one other paper (Ref. 1); that paper was discovered after the present effort was well under way.

THE PROGRAM LOGIC

Virtually every (numerical) computer program can be regarded as a map from mandatory inputs to invariable outputs. The goal of this program is to relax this one-to-one map and provide a complete set of outputs from any sufficient set of inputs. To begin the process, we initially establish the value of all constants and then set the values of all variables to nil.

The heart of the technique is to solve every potentially applicable equation for every variable which occurs in that equation. This should be done analytically, if possible, but we allow for the possibility that iteration, numerical integration, etc., may be required. If, for example, an equation such as

```
W=F1(X,Y,Z)
```

is applicable then we would also solve for

$$X=F2(W,Y,Z),$$

$$Y=F3(W,X,Z),$$

and

$$Z=F4(W,X,Y).$$

Then, for each of these equations we can write statements (LISP is convenient) such as

```
(COND ((AND X Y Z (NOT W))
      ...
(COND ((AND W Y Z (NOT X))
      ...
```

etc. The first condition would be met if and only if X and Y and Z are not nil while W is nil. The second would be met if and only if W and Y and Z are not nil while X is nil. Note that, in non-trivial systems, we may have several equations that yield W as a consequent from varying antecedents so that a specific condition statement for W may never be fulfilled. Once W is produced from any equation, it then becomes available to help produce, say, X or Y or Z from these equations. A rapidly increasing base of known variables results from this approach.

After a condition block has been satisfied the body of the block is filled with various tools that govern the operation of the program. Most important is the evaluation of the consequent which is then added to the accumulating knowledge base; this also guarantees that this block will never again be activated (nor will any other block with the same consequent). All physical units are "conditioned", if necessary, within the block (to allow the user to work in any units that are convenient). An array value is stored to record the order in which a given block was accessed. (This provides a "derivation" of the answer.) Another important function is to set an event-flag to indicate that a new variable has been evaluated. At the beginning of the condition block subroutine, the event-flag is set to nil and, if a new variable is added to the store of known values, the flag is set non-nil. At the end of the condition block subroutine, a non-nil event-flag forces subroutine looping until no new variables are evaluated or until the program terminates. If there are no new values and no termination, further input is requested. (Termination occurs when all variables have non-nil values.)

The mechanics of presenting the flow of information to the user can be handled in many ways. One technique which has proven to be valuable is to present the user with two dynamic menus. One of the menus lists the variables which are presently known (and their numerical values) while the other gives a mouse-sensitive list of variables that have yet to be specified. The "known" menu grows at a very rapid pace because one input often yields many new variable values. The variables which are known, and others derived from them, are removed from the "as yet unspecified" listing on that dynamic menu.

THE HOHMANN TRANSFER

Although page limitiations preclude non-trivial examples, the classic Hohmann transfer is illustrative. This maneuver involves a rocket vehicle leaving a circular orbit by adding an impulsive velocity along the tangent to the orbit, coasting on an ellipse to a higher circular orbit and entering into that orbit via a second tangential impulse.

For circular orbits, if we define the gravitational parameter as

μ, the distance from the attracting primary as R, the magnitude of the velocity vector as V, and if we use subscripts o and f to indicate the initial and final orbits then we have, from the two body problem (Ref. 2),

$$Vo = \sqrt{\mu/Ro} \qquad (1)$$

$$Vf = \sqrt{\mu/Rf} \qquad (2)$$

From these come immediately

$$Ro = \mu/Vo^2 \qquad (3)$$

$$Rf = \mu/Vf^2 \qquad (4)$$

The velocity which must be gained at the two end points of the transfer ellipse are given by

$$\Delta Vo = \sqrt{\mu(2/Ro-1/A)} - Vo \qquad (5)$$

$$\Delta Vf = \sqrt{\mu(2/Rf-1/A)} - Vf \qquad (6)$$

where "A" is the semi-major axis given by

$$A = (Ro+Rf)/2 \qquad (7)$$

From (5), (6), and (7) come the relationships

$$Vo = \sqrt{\mu(2/Ro-1/A)} - \Delta Vo \qquad (8)$$

$$Ro = 2ARo/[A(Vo+\Delta Vo) + \mu] \qquad (9)$$

$$A = \mu Ro/[Ro(Vo+\Delta Vo) - 2\mu] \qquad (10)$$

$$Vf = \sqrt{\mu(2/Rf-1/A)} - \Delta Vf \qquad (11)$$

$$Rf = 2ARf/[A(Vf+\Delta Vf) + \mu] \qquad (12)$$

$$A = \mu Rf/[Rf(Vf+\Delta Vf) - 2\mu] \qquad (13)$$

The "total" velocity increase, V , can be written as the simple sum

$$\Delta V = \Delta Vo + \Delta Vf \qquad (14)$$

which rearranges to give

$$\Delta Vo = \Delta V - \Delta Vf \quad (15)$$

$$\Delta Vf = \Delta V - \Delta Vo \quad (16)$$

If we now introduce M as mass, then the final mass, Mf, is related to the initial mass, Mo , ΔV, and the rocket exhaust velocity, C, as

$$Mf = Mo*\exp(-\Delta V/C) \quad (17)$$

which yields the subsidiary relationships

$$Mo = Mf*\exp(\Delta V/C) \quad (18)$$

$$\Delta V = C*\log(Mo/Mf) \quad (19)$$

$$C = \Delta V/\log(Mo/Mf) \quad (20)$$

Ro, Rf, Vo, Vf, A, Mo, Mf, ΔV, ΔVo, ΔVf and C are thus related by a total of twenty equations. Vo, for example, can come from either equation (1) or equation (8) or from user input. A may be generated from equation (7), equation (10), equation (11) or from input. The only source for Mo is from equation (15) or via input. Etc.

The user may now specify input variables in any order which is convenient. The solution for all variables in all forms from the applicable equations allows a free form input wherein any set of inputs will necessary lead to the output of all unspecified variables.

It is also possible to aid the user by providing help in determining which variables must still be specified to obtain a desired result. By tracing the dependence of the output variable on input variables the user may well recognize a sub-set of inputs that can lead to the required solution. This could allow a solution for smaller problems without having to solve for all possible answers. Graphics aids also have obvious applications.

It must be realized that the above set of equations is so extremely limited that it is doubtful that the corresponding program would hold any users interest for more than a very brief time. To expand to a full design tool requires including options for elliptical initial and target orbits, full three dimensional capability, equations which relate the rocket mass to payload mass, propellant mass, and structural mass, phasing infomation (Gauss' equation requires iteration to isolate the eccentric anomoly), etc. These extensions are presently under way at Marshall Space Flight Center and a useful program is expected within about six months.

CONCLUSIONS

The foregoing has provided only a minimal outline of an important concept in the field of artificial intelligence. Even so, the potential uses of such a system are enormous.

It is apparent that the technique is not limited to mathematics because, for example, it is possible to replace the condition blocks which deal with equations with, for example, chemical synthesis procedures. Once the elementary reactions are available it is possible to begin synthesizing more and more exotic compounds using those building blocks. The system would indicate how to construct extremely complex organic compounds beginning with the elements; help screens could be productively employed to describe the exact laboratory procedures to be followed as well as required equipment.

In another area, it is also possible to "computerize" entire text books and produce programs that would be sold to students as a pony for a given text. With such help it would be impossible to assign a problem to the student which could not be solved using the equations that are supposed to be at hand. The next step would be to combine several texts in differing fields in order to cross-reference knowledge. One would expect that equations and concepts which occur in different fields could be used in unexpected and surprising ways; the end of such an endeavor is not predicable.

REFERENCES

1. Elias, Antonia L. "Knowledge Engineering of the Aircraft Design Process", Chapter 6 of "Knowledge Based Problem Solving", Januez S. Kowalik, Ed., Prentice-Hall, 1987.

2. Burns, Rowland E. "Ascent from the Lunar Surface", NASA TN D-1644, August, 1965. Appendix B.